

Mixed textual/graphical modelling in Papyrus-RT

Ernesto Posse

Zeligsoft

September 13th, 2016



Outline

- Introduction and motivation
- Related work
- Our approach
- Features overview
- Demo

What is Papyrus-RT?

- A new open-source MDE environment for UML-RT
- UML-RT: UML-based language for real-time embedded systems
- Based on Papyrus, an Eclipse-based environment for UML

Collaboration

- CEA List (tooling, validation, import, CDT int.)
- EclipseSource (model compare)
- Ericsson (req., C++ gen. profile, interm. repr.)
- Zeligsoft (codegen, runtime, interm. repr., CDT int., textual, mixed)
- Others

Motivation



Ed Seidewitz
@seidewitz



Follow

@con_sultan @Grady_Booch @siliconglen
@simonbrown @martinfowler Graphical notation
can be powerful, but it is not best for everything.

9:10 AM - 4 Aug 2016

Motivation

- Graphical notations
 - Advantages
 - Clear depiction of relationships and structure
 - Do not require specific textual notations
 - Disadvantages
 - Graphical modelling tools may have UI problems
 - Difficulties with copy/paste, search/replace, diff/merge
 - Storage formats are not meant for human readability
 - Graphical notations are not always adequate

Motivation

- Textual notations
 - Advantages
 - Simpler and more reliable and mature tools (text editors, regex search/replace, diff/merge)
 - Simple storage format
 - More flexible notations
 - Disadvantages
 - Structure may not be as evident as a graphical view

Motivation

- Graphical vs. Textual: false dilemma
- Modelling is about *abstraction* and *representation*
- ... not about (graphical) *notation* and *presentation*
- No commercial tool (that we are aware) supports both

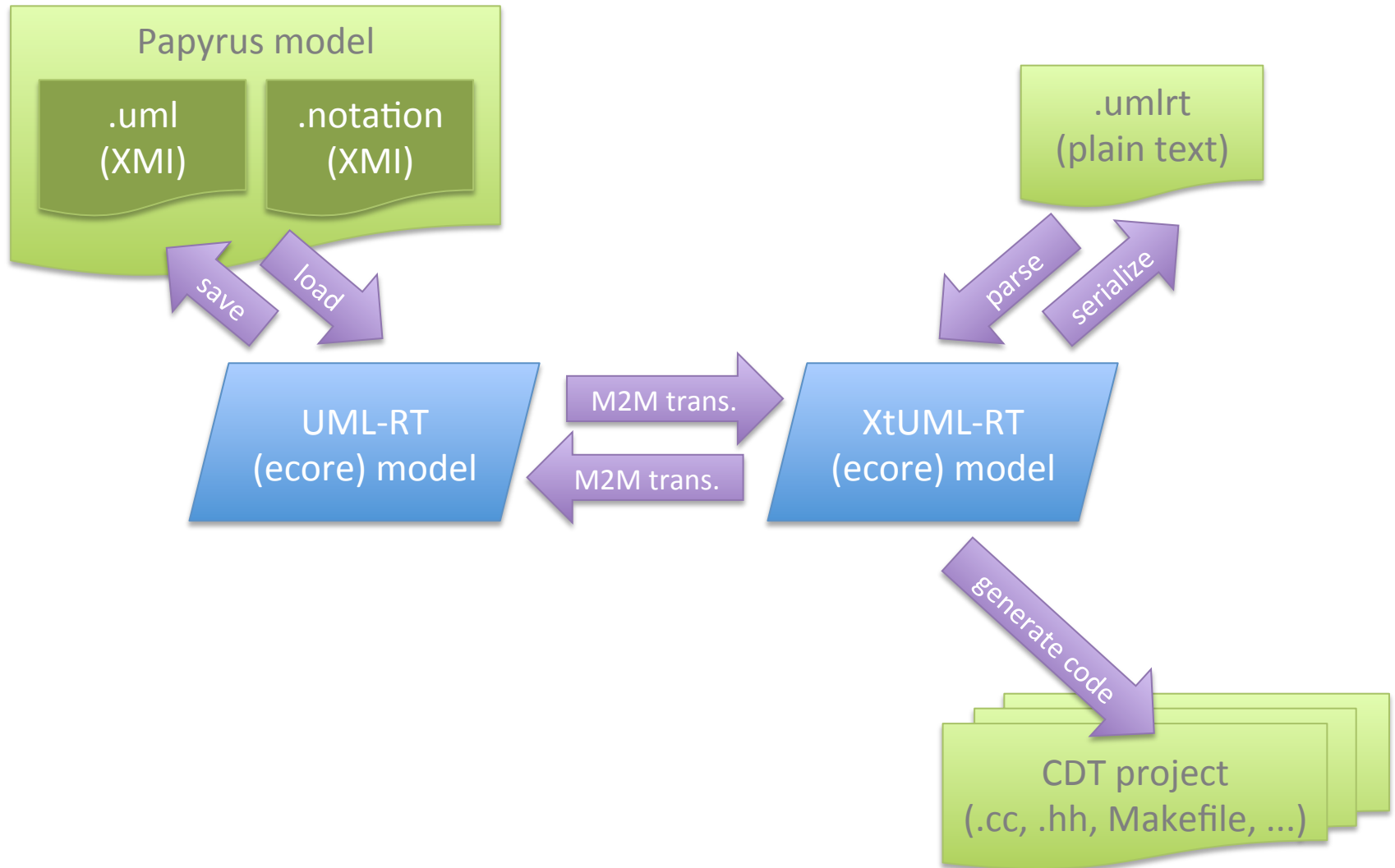
Related work

- [1] Atkinson, C., Gerbig, R. *Harmonizing Textual and Graphical Visualizations of Domain Specific Models*. GMLD 2013.
- [2] Maro, S. *A DSL Supporting Textual and Graphical views*. M.Sc. Thesis. U. Gothenburg. 2014.
- [3] Maro, S., Tichy, M., Gelin, L. *On Integrating Graphical and Textual Editors for a UML Profile Based Domain Specific Language*. SLE 2015.
- [4] Schneider, C. *Integrating Graphical and Textual Modeling*. Diploma Thesis. C. A. U. Kiel. 2011.
- ... several others

Related work

- Projectional editors
 - MVC-like paradigm
 - Central artifact is a model
 - Views are graphical or pseudo-textual (form-based) editors
 - Advantage: simplifies views synchronization
 - Disadvantage: representation is not text (so you need a custom editor and cannot use textual-processing tools)

Our approach



Papyrus-RT features

- Graphical and textual modeling
- Code generation (to C++)
- Run-time system (Linux/gcc, Windows/VisualStudio)
- CDT integration
- RSA model migration
- Model compare/EGit integration

UML-RT

- Concurrent, Object-Oriented, Real-Time systems
- UML with constraints and new concepts
 - Composite Structure Diagrams
 - State Machine Diagrams
- Capsules:
 - (Re)Active classes
 - Interaction via message-passing through ports

Demo

- Simple graphical model
 - Can edit, validate, generate code
- Transform graphical model to text
 - Language: capsules, protocols, classes, SMs, interactions
 - Navigation, folding, content-assist, generate code, CDT integration
- Transform text to graphical
 - Show model structure, Capsule diagram layout

Wrap-up

- Work in progress - feedback wanted
- Advantages of mixed textual/graphical
 - Convenience/user preferences
 - Interoperability, e.g.:
 - PlantUML produces drawings while Papyrus-RT gives you models that can be manipulated (e.g. code generation, validation, analysis, etc.)
- Extensibility (both tooling and codegen)

Links

- Website

<https://www.eclipse.org/papyrus-rt/>

- Wiki

<https://wiki.eclipse.org/Papyrus-RT>

- Bugzilla

<https://bugs.eclipse.org>

- Git/Gerrit

[https://git.eclipse.org/r/papyrus-rt/
org.eclipse.papyrus-rt](https://git.eclipse.org/r/papyrus-rt/org.eclipse.papyrus-rt)